innovaycree.com/blog · documento generado del artículo en línea; la versión vigente es la URL canónica.

SEGURIDAD DE IA · Nº 01 · V1.0

Agentes de IA en producción: qué controles exigir

> Un agente de IA no solo responde: ejecuta acciones con efectos reales sobre sistemas y personas. Antes de dejarlo actuar hay seis controles que un comprador debe exigir y comprobar: permisos mínimos otorgados por herramienta y no por agente, tratamiento de todo contenido leído como entrada no confiable, compuerta humana en las acciones irreversibles, registro que permita reconstruir cualquier decisión, interruptor de parada con modo degradado y topes de gasto y de iteración. Explicamos cada uno, cómo se comprueba y qué falla cuando no está.

Por **David Pinto** — Fundador y arquitecto de la plataforma · **LinkedIn**

Revisión técnica: Innova y Cree

Publicado 17 ago 2026 · 11 min de lectura · PDF

Antes de dejar que un agente de inteligencia artificial actúe sobre sistemas reales hay seis controles que deben estar puestos y ser comprobables: **permisos mínimos** concedidos por herramienta y no por agente, tratamiento de todo el contenido leído como **entrada no confiable**, **compuerta humana** en las acciones irreversibles, **trazabilidad** suficiente para reconstruir cualquier decisión, **interruptor de parada** con modo degradado y **topes** de gasto y de iteración. Ninguno de los seis es una función del modelo: los seis viven en la capa que rodea al modelo, y por eso se pueden exigir por contrato y comprobar en una sesión técnica. Este artículo explica cada control, cómo se verifica y qué falla cuando falta. Complementa las diez preguntas de cómo

evaluar a un proveedor de IA, en particular la que trata sobre qué puede hacer la IA por su cuenta.

Qué distingue a un agente de un asistente que solo responde

Un asistente devuelve texto; el efecto sobre el mundo lo produce después una persona que lee y decide. Un agente invoca herramientas: escribe en una base de datos, envía un mensaje a un cliente, genera un documento, mueve un registro de estado, consulta un sistema externo, a veces cobra. Esa capacidad se llama **agencia**, y cambia por completo el perfil de riesgo, porque el error deja de ser una frase equivocada en pantalla y pasa a ser una acción ejecutada.

OWASP lo recoge en su lista de riesgos para aplicaciones con modelos de lenguaje como *agencia excesiva*: sistemas a los que se concedió más funcionalidad, más permisos o más autonomía de la que la tarea requería[1][2]. La lista específica para aplicaciones con agentes añade los riesgos que solo aparecen cuando hay herramientas, memoria y varios agentes coordinándose[3], y la guía de amenazas y mitigaciones los organiza por patrón de agente, que es la forma útil de leerlos cuando hay que decidir controles[4]. La regla de partida es simple: **la autonomía se concede por tarea y se retira por defecto**, no al revés.

Control 1: permisos mínimos, por herramienta y por alcance de datos

El permiso no se concede al agente sino a cada herramienta que el agente puede invocar, y con el alcance más estrecho que permita cumplir la tarea. Un agente que solo debe consultar el estado de un pedido no necesita permiso de escritura sobre pedidos; uno que envía correos a una lista definida no necesita poder escribir a cualquier dirección.

Tres condiciones hacen que el control sea real. Primera: los parámetros de cada invocación se validan **fuera del modelo**, contra un esquema y contra listas de valores permitidos; el modelo propone, el código dispone. Segunda: la identidad con la que actúa el agente es propia y distinta de la del operador humano, para que el registro distinga quién hizo qué y para que retirarla no requiera tocar cuentas de personas. Tercera: los permisos caducan y se revisan; un permiso concedido para una migración que sigue vivo seis meses después es un hallazgo, no una comodidad.

Conviene decirlo sin rodeos porque es el error más común: **escribir en la instrucción del agente «no hagas X» no es un control de acceso**. Es una preferencia expresada en el mismo canal que un atacante puede manipular. Los límites que importan se aplican donde el modelo no llega.

Control 2: todo lo que el agente lee es entrada no confiable

La inyección indirecta de instrucciones es el riesgo característico de los agentes: el contenido que el sistema lee para trabajar —un correo, una página web, un PDF adjunto, el nombre de un archivo, una fila de una tabla— contiene texto dirigido al modelo. El agente no distingue de forma nativa entre el encargo de su operador y las palabras del documento que está procesando, de modo que un documento puede convertirse en un canal de órdenes.

El tratamiento correcto tiene tres piezas. **Separación**: el contenido recuperado viaja marcado como datos, en un canal distinto del de las instrucciones, y el sistema declara explícitamente que nada de lo recuperado tiene autoridad para cambiar los objetivos. **Neutralización**: se sanea lo que llega —enlaces, marcas, texto oculto, instrucciones aparentes— antes de que entre al contexto. **Contención**: aunque la inyección funcione, el daño queda acotado porque los permisos son mínimos y las acciones sensibles requieren aprobación. Esta última pieza es la única que resiste cuando las dos primeras fallan, y por eso el diseño supone que fallarán. La matriz de

MITRE ATLAS documenta estas técnicas contra sistemas de IA y es una buena base para redactar los escenarios de prueba[5].

Control 3: compuerta humana en lo irreversible

No todas las acciones necesitan aprobación, pero algunas no deberían ejecutarse jamás sin ella. El criterio no es la importancia sino la **reversibilidad**: si deshacer la acción es imposible, costoso o visible para un tercero, hay compuerta.

La clasificación práctica tiene tres niveles. Acciones **libres**: lectura, búsqueda, redacción de un borrador que nadie recibe. Acciones **con registro**: escrituras internas reversibles, cambios de estado que se pueden revertir dejando rastro. Acciones **con aprobación**: todo lo que sale de la organización —mensajes a clientes, publicaciones, presentaciones ante terceros—, todo lo que toca dinero y todo lo que borra. La compuerta debe mostrar a quien aprueba exactamente lo que se va a ejecutar, no un resumen; aprobar un resumen es aprobar a ciegas.

El Reglamento europeo de inteligencia artificial convierte esta práctica en obligación para los sistemas de alto riesgo: exige medidas de supervisión humana que permitan a la persona comprender, vigilar e interrumpir el funcionamiento del sistema[6]. Donde ese reglamento no aplica, el control sigue siendo el mismo; lo que cambia es quién lo exige. El cuadro de qué norma rige en cada país está en la tabla de marco legal del artículo pilar.

Control 4: trazabilidad que permita reconstruir la decisión

Un registro sirve cuando permite responder, meses después, por qué el agente hizo lo que hizo. Eso exige guardar, con marca de tiempo y por cada paso: quién originó la solicitud, qué contexto se recuperó y de dónde, qué herramienta se invocó con qué parámetros, qué devolvió, qué decidió el agente a continuación y quién aprobó cuando hubo aprobación.

Falta casi siempre un dato, y es el que vuelve inútil al resto: **la versión del modelo y de la instrucción vigentes en ese momento**. Sin ella, el registro cuenta qué pasó pero no permite explicarlo, porque el sistema que respondió aquel día ya no existe. Un alias comercial de modelo, que el proveedor mueve cuando publica una versión nueva, produce exactamente ese vacío; por eso el inventario debe declarar la versión fijada, como se explica en el inventario de componentes de IA.

La función GOVERN del marco de gestión de riesgo del NIST pide precisamente esto: responsabilidades asignadas, decisiones documentadas y evidencia conservada, no confianza en la memoria de quien operaba[7].

Control 5: interruptor de parada y modo degradado

Un interruptor de parada no es apagar el servidor. Es poder desactivar **una herramienta o un agente concreto** sin derribar el servicio, y que el sistema quede en un estado útil y seguro: consulta y lectura disponibles, acciones suspendidas. Tres exigencias lo convierten en un control real.

Primera: el mando debe estar al alcance del cliente, no solo del proveedor, y debe surtir efecto en minutos, no en el plazo de un ticket. Segunda: el sistema debe declarar explícitamente que está degradado, en la interfaz y en el registro; un agente que deja de actuar en silencio es peor que uno detenido con aviso, porque nadie sabe qué quedó a medias. Tercera: debe existir un procedimiento para las acciones en vuelo cuando se acciona el interruptor —abortarlas, completarlas o dejarlas en cola— y el cliente debe saber cuál es. Un interruptor que nunca se ejercitó en simulacro es una casilla de un documento, no un control.

Control 6: topes de gasto, de iteración y de alcance

Un agente que puede llamarse a sí mismo puede entrar en bucle, y un bucle con herramientas de pago es una factura. Los topes son aburridos y son los que evitan

los incidentes más caros: número máximo de pasos por tarea, número máximo de invocaciones por herramienta, presupuesto máximo por tarea y por día, tiempo máximo de ejecución. Todos aplicados **en código**, no como recomendación en la instrucción.

A los topes conviene añadirles alarmas por umbral y un comportamiento definido al agotarlos: detenerse y avisar, nunca continuar en silencio. Y una regla de alcance: un agente no debería poder ampliar por su cuenta el conjunto de herramientas al que tiene acceso, ni descubrir nuevas en tiempo de ejecución, sin que ese cambio pase por el mismo control que la concesión inicial.

Dos cicatrices propias, contadas en genérico

Los dos fallos siguientes ocurrieron en nuestra operación, se detectaron midiendo y se corrigieron. Los contamos porque describen categorías enteras de error que ningún control genérico atrapa si no se busca.

El agente prometía algo que un filtro descartaba. Un asistente en canal de mensajería anunciaba al cliente que haría llegar sus archivos de voz a la persona responsable. Un filtro por tipo de archivo, aguas abajo, los descartaba antes del envío. Nada en el sistema estaba «caído»: el aviso se enviaba, el registro decía «notificación enviada» y el panel estaba en verde. La lección es que **el aviso de una acción no es la acción**, y que la verificación debe alcanzar el efecto final —¿llegó el archivo?—, no el paso intermedio que el sistema controla. Desde entonces, toda promesa que un agente pueda formular tiene que corresponder a una capacidad declarada y probada de extremo a extremo; si el filtro descarta algo, el agente no puede ofrecerlo.

El transcriptor inventaba sobre silencio. Un componente de reconocimiento de voz devolvía una frase de cortesía perfectamente plausible al procesar audio sin habla. Sin una guardia previa que exigiera señal de voz, esa frase entraba al flujo como si

fuera lo que el cliente había dicho, y a partir de ahí el agente razonaba sobre algo que nadie dijo. El NIST llama a esto confabulación y lo enumera entre los riesgos propios de la IA generativa[8]. La lección: **toda entrada automática necesita una guardia de plausibilidad antes de convertirse en un hecho** dentro del sistema, y las salidas de un componente de IA que alimentan a otro son el punto donde un error se multiplica.

Los seis controles y su evidencia

#	CONTROL	QUÉ PEDIR COMO EVIDENCIA
1	Permisos mínimos por herramienta	Lista de herramientas con su alcance, el esquema de validación de parámetros y la identidad propia del agente
2	Entrada no confiable	Descripción de la separación entre datos e instrucciones y el resultado de pruebas de inyección indirecta
3	Compuerta humana	Clasificación de acciones por reversibilidad y una captura real de la pantalla de aprobación
4	Trazabilidad	Un registro de ejemplo con parámetros, resultado, aprobador y versión de modelo e instrucción
5	Interruptor y modo degradado	Quién puede accionarlo, en cuánto tiempo surte efecto y el acta del último simulacro
6	Topes de gasto e iteración	Los límites configurados, dónde se aplican y qué ocurre al alcanzarlos

los seis controles exigibles a un agente de IA en producción y la evidencia que los acredita.

Cómo encaja esto en el resto del expediente

Estos controles no viven solos. La **política de IA responsable** del proveedor debe describir quién autoriza una ampliación de autonomía y qué se re-mide antes de

concederla; el detalle está en qué debe decir una política de IA responsable. El **inventario de componentes** debe declarar qué modelos y qué terceros participan en cada herramienta que el agente invoca, con su país y su versión fijada[1]; ese documento es el AI-BOM. Y la **medición** de exactitud debe cubrir también las tareas del agente, no solo las respuestas de texto: cómo se mide está en cómo medimos las alucinaciones de una IA en producción y el método de verificación de lo que el agente afirma, en cómo se verifica una cita generada por IA.

Un comprador que pide los seis controles con su evidencia obtiene, en una sola sesión, una imagen fiel de la madurez del proveedor. Si quiere revisar un caso concreto, puede plantearnos el escenario.

1. OWASP GenAI Security Project, *LLM Top 10 — 2026*; incluye inyección de instrucciones, agencia excesiva y cadena de suministro. Consultado el 17 de agosto de 2026. ↩↩

2. OWASP, *Top 10 for Large Language Model Applications* (página del proyecto). Consultado el 17 de agosto de 2026. ↩

3. OWASP GenAI Security Project, *Top 10 for Agentic Applications (2026)*. Consultado el 17 de agosto de 2026. ↩

4. OWASP GenAI Security Project, *Agentic AI: Threats and Mitigations*. Consultado el 17 de agosto de 2026. ↩

5. MITRE ATLAS, matriz de tácticas y técnicas adversarias contra sistemas de inteligencia artificial. Consultado el 17 de agosto de 2026. ↩

6. Reglamento (UE) 2024/1689, artículo 14 (supervisión humana). Consultado en EUR-Lex el 17 de agosto de 2026. ↩

7. NIST, *AI Risk Management Framework 1.0*, función GOVERN. Consultado el 17 de agosto de 2026. ↩

8. NIST, *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile* (NIST AI 600-1); la confabulación figura entre los riesgos propios de la IA generativa. Consultado el 17 de agosto de 2026. ↩

Puntos clave

01 Un agente se distingue por la agencia: ejecuta acciones con efecto real, y por eso los controles se diseñan antes del despliegue, no después del primer incidente.

02 Los permisos se conceden por herramienta y por alcance de datos, con parámetros validados fuera del modelo; una instrucción en el prompt no es un control de acceso.

03 Todo contenido que el agente lee para trabajar es entrada no confiable: la inyección indirecta llega en documentos, correos y páginas, no solo en lo que escribe el usuario.

04 Las acciones irreversibles pasan por compuerta humana, y el registro debe permitir reconstruir cualquier decisión con la versión de modelo e instrucción vigentes.

05 El interruptor de parada, el modo degradado y los topes de gasto e iteración se prueban en simulacro; si nunca se ejercitaron, no existen.

Preguntas frecuentes

¿Qué diferencia a un agente de un asistente que solo responde?

¿Qué es la inyección indirecta de instrucciones?

¿Basta con pedirle al agente en su instrucción que no haga cosas peligrosas?

¿Qué debe registrar un agente para que una acción sea auditable?

¿Un interruptor de parada es apagar el servidor?

Referencias

[1] OWASP GenAI — LLM Top 10 (edición 2026): inyección de instrucciones, agencia excesiva y cadena de suministro. https://genai.owasp.org/resource/owasp-genai-llm-top-10-2026/. Consultado el 17 de agosto de 2026. NORMA

[2] OWASP GenAI — Top 10 for Agentic Applications (2026). https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/. Consultado el 17 de agosto de 2026. NORMA

[3] OWASP GenAI — Agentic AI: Threats and Mitigations (taxonomía de amenazas y controles por patrón de agente). https://genai.owasp.org/resource/agentic-ai-threats-and-mitigations/. Consultado el 17 de agosto de 2026. NORMA

[4] OWASP — Top 10 for Large Language Model Applications (página del proyecto). https://owasp.org/www-project-top-10-for-large-language-model-applications/. Consultado el 17 de agosto de 2026. NORMA

[5] MITRE ATLAS — matriz de tácticas y técnicas adversarias contra sistemas de inteligencia artificial. https://atlas.mitre.org/. Consultado el 17 de agosto de 2026. NORMA

[6] NIST AI Risk Management Framework 1.0 (funciones GOVERN y MANAGE). https://www.nist.gov/itl/ai-risk-management-framework. Consultado el 17 de agosto de 2026. NORMA

[7] NIST AI 600-1 — Generative AI Profile: riesgos propios de la IA generativa, incluida la confabulación. https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf. Consultado el 17 de agosto de 2026. NORMA

[8] Reglamento (UE) 2024/1689 de inteligencia artificial — artículo 14, supervisión humana de los sistemas de alto riesgo. https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A32024R1689. Consultado el 17 de agosto de 2026. LEY

Citar este artículo

APA

```
Pinto, D. (2026, 17 de agosto). Agentes de IA en producción: qué controles exigir (v1.0). Blog técnico de Innova y Cree. https://innovaycree.com/blog/seguridad-ia/agentes-ia-produccion-controles.html
```

BIBTEX

```
@misc{pinto2026agentes,
  author       = {Pinto, David},
  title        = {Agentes de IA en producción: qué controles exigir},
  howpublished = {Blog técnico de Innova y Cree},
  year         = {2026},
  month        = {ago},
  note         = {v1.0, revisado 2026-08-17},
  url          = {https://innovaycree.com/blog/seguridad-ia/agentes-ia-produccion-controles.html}
}
```

v1.0 · 17 ago 2026

▶ HISTORIAL DE VERSIONES

SOBRE EL AUTOR

David Pinto · Fundador y arquitecto de la plataforma

Diseña la arquitectura de IA verificable de Innova y Cree: RAG con cita verificada contra la fuente oficial, benchmark anti-alucinación público y plataformas en producción para sectores regulados de habla hispana.

LinkedIn · Todos sus artículos